DOCUMENT RESUME

ED 246 875 IR 011 207

AUTHOR Komoski, P. Kenneth
TITLE Educational Software and Other Learning Materials: A Look at the Differences.
PUB DATE 23 Apr 84
NOTE 29p.; Paper presented at the Annual Meeting of the American Educational Research Association (New Orleans, LA, April 23, 1984).
PUB TYPE Viewpoints (120) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS *Autoinstructional Aids; *Computer Assisted Instruction; *Computer Software; *Feedback; *Instructional Materials; Interaction; *Intermode Differences; Media Selection; Parent Role; Socioeconomic Influences; Teacher Role; Teaching Methods; Technological Advancement

ABSTRACT

Factors contributing to the uniqueness of educational software are analyzed from contextual and textual perspectives. Contextual (or external) technological, economic, and social factors are examined briefly, and are identified as the dominant factors in the future evolution of educational software. Feedback and interactiveness, two significant textual or internal factors that constitute major differences between educational software and other learning materials, are discussed within a framework of a theoretical learning materials continuum. These factors are identified as contributing to the development of "instructuring" software, which enables learners to learn independently. Other types of structured software requiring a different type of teacher involvement than that required by traditional instructional materials are also discussed. Questions are raised as to whether and how contextual and textual factors (especially the potential development and use of feedback-based, interactive, instructuring software) may affect current and future educational practices and policies, and some possible effects on the curriculum and on teachers' changing roles are discussed. The role of the home market is stressed along with the need for educator-parent cooperation to ensure a more integrated approach to in-school and at-home use of educational computing. Six references are listed. (Author/LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED246875

EDUCATIONAL SOFTWARE AND OTHER LEARNING MATERIALS: A LOOK AT THE DIFFERENCES

by

P. Kenneth Komoski
Executive Director
EPIE Institute

presented in Symposium

Educational Computer Software: Issues and Trends

at

The American Educational Research Association

1984 Annual Meeting

New Orleans, Louisiana

April 23, 1984

© 1984 EPIE Institute

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

P. Kenneth Komoski

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR011207

ABSTRACT

Factors contributing to the uniqueness of educational software are analyzed from two perspectives, "contextual" and "textual." Contextual (external) technological, economic, and social factors are examined briefly, and are identified as the dominant factors in the future evolution of educational software. Two significant textual (internal) factors that constitute major differences between educational software and other learning materials are identified and discussed within a framework of a theoretical Learning Materials Continuum.

The two unique internal factors of educational software examined are "feedback" and "interactiveness." It is maintained that their uniqueness stems from the "cybernetic core" of computer technology and from evolving instructional technology. Thus, these unique characteristics of educational software are not unique to educational software, per se, but are the two characteristics that clearly set educational software apart from other noncomputer-based learning materials. These factors are identified as contributing to the development of a type of software identified as instructuring: software that is capable of enabling learners to learn on their own with little or no teacher intervention.

Other types of (structured) software requiring a different kind of teacher involvement than required when teaching with traditional instructional materials (i.e., textbooks) are also identified and discussed. The discrepancy between the types of software that now dominates the school and home education market and the potential types and uses of software that may be developed is illustrated by the fact that EPIE Institute's evaluators have found that only three or four out of every 100 evaluated are of excellent instructional and technical quality.

Questions are raised as to whether, and how, the contextual and textual factors discussed in this paper (especially the potential development and use of feedback-based, interactive, instructuring software) may effect current and future educational practices and policies. Some possible effects on the curriculum and on teachers' changing roles are discussed. The important role of the home market is stressed along with the need for educator-parent cooperation to ensure a more integrated approach to in-school and at-home use of educational computing.

EDUCATIONAL SOFTWARE AND OTHER LEARNING MATERIALS: A LOOK AT THE DIFFERENCES

P. Kenneth Komoski

INTRODUCTION

The assumption underlying this symposium is that in a number of ways educational software represents something of a unique phenomenon in the field of learning materials. Just how unique a phenomenon this new type of learning material turns out to be, and whether this uniqueness will have a positive, a negative, or a nonsignificant effect on the future of teaching and learning remains to be seen. But one thing seems clear: increasing amounts of educational software are going to be produced for use in the growing number of computers in schools and in homes.

Because of this, educational practioners and policy-makers need to develop an understanding of educational software vis-a-vis the use of other types of

learning materials and activities. They need to do this in order to make the most effective use of software in relation to other more familiar materials such as textbooks, films, etc. This need is underscored by the fact that this year — as popular as software is — school sales of software will total only about $40 million of the $1.2 billion being spent on other types of learning materials by schools.

As the current expenditure for software grows and software continues to make larger and larger demands on that consistently small percentage of local school budgets (about 1%) spent on learning materials each year, an understanding of the role of software vis-a-vis other materials will become increasingly important for practioners and policy makers. In addition, such an understanding also needs to be developed by parents as well, given the very active role they are playing in the educational computing marketplace.

CONTEXTUAL and TEXTUAL PERSPECTIVES

In an effort to develop a basis for such an understanding of software in relation to other learning materials and their roles in teaching and learning, I propose to examine the phenomenon of educational software from two perspectives. Both of these perspectives demonstrate different ways in which educational software is unique among learning materials.

The first of the two perspectives may be described as contextual. By contextual I mean the social, economic, and technological factors that are contributing to the uniqueness of educational software in comparison to other types of learning materials. Because such factors are external to the products themselves, but help to shape them, "contextual" seems an appropriate way to categorize them.

The second perspective may be described as textual. By textual I mean those internal characteristics of computer software that — somehow or other — are related directly and functionally to the teaching-learning "text" with which the learner interacts in the course of using the software. Of these textual factors I shall concentrate on one in particular that I feel helps to set educational software apart from other learning materials.

Although it is probably true that the large, contextual (social, economic, and technological) factors that have caused educational software to happen in the first place will continue to be the overriding influences in shaping software in the future, it is my intention here to deal with these factors only long enough to demonstrate their importance. I shall then turn to the textual or internal characteristics of educational software that are (1) shared in common with other types of learning materials, and (2) make educational computer software different from those more commonly used learning materials. As an aid to understanding these similarities and differences, I shall employ a theoretical Learning Materials Continuum. Through the use of this Continuum it will be demonstrated that most current educational software is not living up to the educational potential inherent in the two technologies (i.e., computer technology and instructional design technology) that have spawned it.

Before ending I will briefly examine the problems of effectively integrating educational software into the overall learning experiences of students, given the present patterns of materials' use in schools. And I will end by emphasizing the role that school and home (especially the latter) consumers will inevitably play in shaping the quality of educational software in a contextual climate that will be increasingly influenced by the societal acceptance, the economic

profitability, and the technological capability of educational software. In this regard I will urge the forming of educator-parent alliances to help develo more discriminating practices of purchase and use of educational software in schools and homes.

CONTEXTUAL FACTORS

Germinal among the contextual factors that are influencing educational software has been the low-cost computer hardware to which all software (educational and otherwise) is inextricably related. The heart of that hardware, the silicon chip, represents the technological "egg" from which the other important economi and social contextual effects that are influencing educational software have be hatched. Thus, once again (as was the case with the automobile, television, an nuclear power), a new technology is precipitating important economic and social effects.

Just how pervasive and long-lived these effects will be on educational practice that remain essentially unchanged despite 30 years of technological innovations in schools is difficult to predict. Nevertheless, there are new, and undeniabl powerful technological, economic, and social factors impinging on those seeming intractable practices and policies. Technologically, there is, of course, the low-cost (and increasingly lower- and lower-cost) computer itself. But the computer, by itself (along with its symbiotic sister, software), would not be receiving anything like the attention it is currently getting from educational practioners and policy-makers if it were not for the surprisingly pervasive economic and social impact of this new technology that is being felt by parents. More than any other factor it is parental awareness of the economic-social climate being generated by the computer that is making educational practioners

and policy-makers pay attention to the educational implications of this new technology — whether they want to or not.

But the economic factor that is clearly operating is that the low-cost computer has, in fact, become low enough in cost to become a mass-market item with broad-based business, household, and educational applications. As a result, the machinery of mass-marketing has all but guaranteed the current computer frenzy among parents. Thus many educators feel they are being forced to introduce computing to schools because of a totally unique, unprecedented parental response to a technology. Parents who in the past have only tolerated their school's use of TV and similar technological innovations are today demanding that their schools use computers.

Nor does this parental response stop with a request or a demand. In many communities in which school budgets are at their lowest in modern memory, parents themselves are raising the funds necessary to assure that there are, indeed, computers in their schools. The result is that market researchers report that last year parent computer activitists were responsible for the purchase of about one out of every five computers being used in schools. However, this phenomenon no longer needs to continue because parents, having made their point, have witnessed a doubling of school computer purchases during the current school year.

But as significant as this in-school parental influence has been, perhaps of even greater significance may be the amount of money parents are spending on educational software for their children's use at home. (Informal field research by this writer based on questions put to large groups of computer-owning parents in a dozen states indicates that parents of school-age children who own a home

computer are spending an average of $300 this year per child on educational software for home use.) When the parent expenditure of $300 a year is compared to the approximately two dollars per child being spent for software this year by schools with computers, the implications for local school districts may be considerable. Nationally, through 1984, this will translate into about $75-100 million in home educational software sales as against the forty or so million dollars in school software sales. If this rate of expenditure on educational software for home use grows as expected, the home market for educational software could come to outstrip the school market to such a degree that school purchasers might receive scant attention from an industry which once depended almost entirely on school purchases for its livelihood.

One result of all this parental interest in computer learning at home as well as in school has been a broadly expanded array of educational software companies. (Due in large part to potential home sales of educational software, the number of education companies has increased during the last five years from about one hundred to more than five times that number.) Just how many of the hundreds of new companies will survive in the increasingly competitive, rapidly expanding home/school marketplace is one more question that adds to the contextual complexity of educational software. One thing does seem likely, though: The educational marketplace, once synonymous with "the school market," is being expanded, perhaps even overwhelmed, by "the home market" for educational products.

I point out these phenomena only as a way of saying that the long-range effects of the still-developing technological-economic-social context being generated by the computer on established educational practices and policies are difficult to predict.

TEXTUAL FACTORS OF SOFTWARE AND OTHER LEARNING MATERIALS

Having said this much — but hardly all that might be said — about the major contextual factors shaping educational software, I want now to turn my focus to the internal, textual characteristics of that software and, in particular, on those attributes that set it apart from other learning materials. It is this uniqueness that needs to be better understood if educators, parents, and students are to effectively relate learning with computers to other types of learning materials and experiences, and if educational software is to be used effectively to improve teaching and learning.

In order to identify and effectively discuss those characteristics of educational software that give it its unique character, it will be helpful to note those characteristics of educational software that are shared with other types of learning materials. And in order to do this I want to employ a schema which was developed some time ago at EPIE Institute as a way of organizing and clarifying functional differences among the full range and variety of learning materials used by teachers and students. (EPIE 1975)

This schema is presented here as a Learning Materials Continuum along which every type of learning material may be arranged according to its function vis-a-vis the role and decision-making of the teacher and according to its internal complexity and the intentionality of its educational and instructional design.

The Learning Materials Continuum

Types of Materials	Astructional materials	Structured materials	Instructional materials	Instructuring materials
	(least complex)	(increasingly complex)		(most complex)
	(Nonintentional--no inherent or intentional instructional design. Teacher must make all instructional design decisions.)	(Some potential but nonintentional design inherent in the structure or organization of objects or content. Some inadvertent decisions built into structure that may influence instructional design decisions made by teacher/learner.)	(Some intentional instructional design. Increasing number of intentional instructional design decisions made by material's developer rather than by teacher/learner.)	(Highly intentional--all but the most trivial curriculum and instruction decisions made by material's developer. Few, if any, left to teacher/learner.)

At the left side of the Continuum we find materials that, although they are often used for teaching/learning purposes, were never intentionally designed to facilitate learning. Such materials, because they are devoid of educational interaction and instructional design, are labeled "astructional." Astructional materials range from available "found" objects in the environment to printed words and pictures.

When teaching with such nonintentional, astructional, materials teachers must deal — intuitively or consciously — with a myriad of curriculum and instructional design decisions, many of which are highly complex and demanding. Despite these requirements, some teachers succeed beautifully in accomplishing important intended learning outcomes by using such common materials as chairs, the walls of a room, seashells, a leaf, a photograph, a television drama, an animal, an automobile — even a computer which, when devoid of application software that structures its functioning in a specific way, may be viewed as an astructional material. Teachers who use astructional materials have both the professional freedom and the professional responsibility to make every single curriculum and instruction decision needed to bring about the intended learnings they have decided upon.

Structured materials (the second category on the left of the Continuum) range from organized collections of objects, such as mineral or art collections, newspaper clippings, a series of documentary films, The New York Times Index, to electronic databases that have no specific built-in educational intention — no explicit or implicit instructional design goals and objectives. However, teachers frequently use such structured materials as aids in pre-structuring content for student examination. Also, the logical or chronological structure of

such material often helps a teacher to organize instructional activities and learning experiences deemed relevant to achieving a school's intended learning outcomes.

Computer software that may be categorized as structured materials are word-processing software, database software, and searchable databases, as well as a variety of computer languages (LOGO, perhaps, being the most obvious). The major educational potential of such software seems to lie in its ability to provide structured electronic environments within which learners may experiment with new ways of writing (in the case of word processing), manipulating information (databases and their management), and using the structure of a computer language (LOGO) to discover how to control a computer.

Teachers employing such nonintentional, but structured materials will find that, while they still have the freedom to make critically important decisions of educational intentionality and instructional design, such materials do contain a level of inherent organization and structure that may either enhance or limit the teacher's ability to achieve certain types of educational outcomes.

When we move further along the Continuum toward explicitly intentional learning materials we find those instructional materials commonly called textbooks, workbooks, film strips, teaching films, multi-media kits, etc. These materials have been intentionally created for use by teachers and learners to bring about intended educational goals and specific instructional objectives but that do not state or imply that. When used by learners, the material is capable of enabling individual learners, on their own, to "instructure" or self-instruct themselves in the acquisition of a new skill, or a body of knowledge, or a new behavior by simply using such a material. Such instructional materials are designed with the

expectation of a teacher's involvement. With an instructional material, the task of "instructuring" learning is assumed to be the teacher's job.

In order to help the teacher do this, many textbooks, workbooks, filmstrip series, teaching films, training systems, and increasingly, even instructional computer software systems are accompanied by teachers' guides, designed to help teachers to achieve the intentional goals and objectives that have been built into such materials by their developers. It should be recognized that such guides and the materials they accompany are as good or poor as the quality of the curriculum and instructional design decisions that have been made by those developers. In short, schools that purchase and teachers who use such materials have delegated all of the most important curriculum and instructional design decisions to the material's developers — a fact of curriculum life that frequently eludes many educators and school policy-makers.

Nevertheless, with a carefully selected instructional material that fits a school's curriculum, its teachers' instructional requirements, and its learners' capabilities and motivation, teachers may still have a feeling of instructional freedom and responsibility. This feeling may be quite justified when the design decisions delegated to the material's developer are compatible with the decisions that the teacher would have made had she/he been involved in the material's design. Thus, by using such materials they also gain freedom <u>from</u> the responsibility of making all curriculum and instructional design decisions.

However, when the material selection process has failed to produce an adequate fit between a material, a teacher, and a learner, instructional design decisions made by the material's developer are frequently disagreed with or ignored by the teacher. In such cases, a teacher may relegate the material to the role of

"structured" content to which he or she must bring the "proper" instructional design decisions to meet specific needs (e.g., rearranging a text's organization of chapters, or taking a different approach to teaching its content). Microcomputer software of various sorts may be classified as instructional, especially drill-and-practice and tutorial software (the teachers' guides of which will be examined in another paper during this Symposium).

The final and most complex category, on the far right of the Continuum, is reserved for what we have characterized as those highly intentional, self-instructural, or instructuring materials, for which the medium of computer software seems uniquely suited.

An instructuring material is one that has been so completely designed and adaptively shaped by its developer that every possible curriculum and instructional design decision, including diagnosis, prescription, remedial sequences, alternative approaches to the learning, and assessment of intended learning outcomes are all fully built into and mediated by the material itself, with minimal teacher involvement and practically no critical teacher decision-making beyond the management decision that a particular material is appropriate for that particular learner. If such a material is, in fact, truly appropriate for a particular learner, it is the interaction between the learner and the material that produces a material-mediated "instructuring" of the material's intended learning outcomes resulting from the learner's interaction with the material. In short, the material is the teacher (or more correctly put: the developer who has created the material is the teacher). And it is the material (i.e., the material's developer) that has taken on the responsibility of engaging the learner's attention and energy in the learning task.

Although it is hard to imagine an instructuring material appearing in textbook form or in the form of other familiar learning materials (over the years there have been attempts to market such "teacher-proof" materials with mixed success), one could make a strong case that instructuring materials have not been truly feasible before the advent of the computer with its interactive, feedback-based, multi-sensory teaching-learning capabilities. As true as this may be, however, once the microcomputer is effectively and economically interfaced with other technologies like the videodisc, we can expect to see an increasing flow of instructuring materials entering the educational marketplace. And as such materials begin to be used, it will be well to remember that their use will be accompanied by greater and greater delegation of curriculum decisions to those who develop such materials.

If we were to arrange a frequency distribution of materials currently in day-to-day use in American classrooms along this Learning Materials Continuum, the greatest density of the distribution would be found from just to the right of center of the Continuum at the category labeled "Instructional." This, of course, reflects the traditional domination of the textbook, but it is also true for computer software as well, except for the increasing use of structured materials such as word-processing, database management, and computer language software. But overwhelmingly, today's commonly used materials are instructional, intentional, and teacher-assisted, with much less use of nonintentional, astructional and structured materials that, by their nonintentional nature, would need much more than teacher assistance to be used effectively.

While the terms employed on this Learning Materials Continuum may not be the most linguistically felicitous, they may be defended as being a good deal more descriptive of some functionally important differences among the great variety of

learning materials used in today's schools than those terms commonly used to describe such materials. I refer to such terms as "textbook," "workbook," "audiovisual (or the alternative 'nonprint') materials," "ITV" (instructional television), and now "computer software" or just "software." Such terms deal only with the most obvious attribute of a learning material: its medium. Such terms fail to communicate anything about the internal (textual) and functional structure of a given material.

For instance, the term "textbook" by itself tells us nothing more than that such a material contains text. Yet, as anyone who has spent time analyzing textbooks knows, there is great variance within such materials from the standpoint of instructional design and the ability of the material to communicate effectively with learners. Yet, to many, a textbook is a textbook.

If anything, the term "audiovisual" and its alternative, "nonprint material" is even less functionally descriptive than "textbook." And the frequently-used "supplementary materials" is equally unhelpful. In fact, with the exception of the term "textbook" (which usually refers to a material that can be characterized as an intentionally instructional material), other terms such as "audiovisual," "supplementary," "nonprint," and "nontextbook-print materials" are commonly used to describe materials that may, in fact, range across the full scope of the Continuum. And, as we have seen, as a descriptive term, "computer software" is no exception.

Nonetheless, it is interesting to note that for some reason computer software has brought a set of instructionally relevant terms into the world of learning materials that never have been widely used to describe more familiar learning materials. Terms like "drill-and-practice," "tutorial," "simulation," and

"gaming" describe well-known learning activities and strategies. But we seldom hear of these strategies used to describe textbooks or audiovisual materials. Of course, "workbooks" are actually drill-and-practice books. But somehow they seem to be thought of as the "work" part of common learning materials, as though using a textbook or other materials requires no work. The terms used to describe instuctional computer software leave little doubt that some sort of purposeful activity (i.e., work) on the part of the learner is involved in their use.

But, as we have seen, not all computer software used in schools is instructional. Some may be quite as "astructional" or educationally unintentional as other types of "found" materials used by an imaginative teacher to accomplish a variety of teacher-decided learning outcomes. Other software such as structured computer languages — LOGO being the most common example — are being used by teachers and learners to accomplish a wide range of unspecified, if not always unintended learning outcomes: exploring logical, cognitive, and/or esthetic relationships, solving a variety of problems, expressing verbal, graphic, and musical artistry.

Still other, explicitly intentional software, such as instructional drill-and-practice and especially tutorial programs that simulate good, adaptive teaching, can be far superior to noncomputerized instructional programming. But this tends to be true only when the interactive, adaptive, feedback, and record-keeping capabilities of the computer are fully and appropriately exploited. When these capabilities are effectively exploited in software that incorporates the sort of complex behavior demonstrated by the capable teacher, it should be noted that the software's teaching capability is the result of that software's developer having known enough about teaching to be able to build a teaching capability into the software. As a result, with an <u>instructuring</u> material, instead of the learner's behavior being <u>instructured</u> as a result of the traditional triad of learner-

teacher-material, the operative interactions are between the learner and the material.

Having made note of the broad range of capabilities — from _astructional_ to _instructuring_ — possible for computer software as a learning material, we can see that such software has much in common with the many traditional learning materials that also range across the Continuum. However, computer software does seem to have a unique _instructuring_ potential. In addition to its unique feedback-facilitating capacity, this unique _instructuring_ potential is, I believe, also due to the unique capability of digital computerized communications to make possible very flexible, _interactive_ communications: via the printed word one minute, through static and animated graphics another (and, as needed, being able to supplement words and pictures with audio), and finally, being able to do all of this with the sort of adaptive, diagnostic, prescriptive capability associated with a good human instructor.

So it seems, then, that although computer software shares many pedagogical strategies with other, more familiar, learning materials and media, its uniqueness stems from its capacity for carrying out those strategies more flexibly, more interactively, and, potentially, more effectively. For instance, a well-designed textbook can provide a learner with searchable, randomly accessible information. But through the effective use of feedback, well-designed software can provide such information in response to an ongoing objective analysis of the learner's performance and needs. And although any printed learning material can employ text-plus-illustrations to help learners understand, and films and television can provide animated instruction, only computer software can employ printed text and still and animated sequences with such great flexibility, selectivity, adaptability, and ease.

This capability of employing text, animated pictures and sound selectively and interactively to ensure learning effectiveness is perhaps best demonstrated in simulation software designed to help a student to acquire (instructure) new skills or knowledge by interacting directly with a simulated environment available through the software designed specifically to make such learning possible. However, even when it comes to more traditional teaching/learning strategies, such as drill-and-practice, didactic prose, testing and remediation, if the software in question effectively records, analyzes, and provides feedback to the learner, such educational software can also be superior to textbook exercises, workbooks, and other traditional learning materials. It should be noted that, for the learning of certain skills, such software is not necessarily superior to teacher-designed activities involving practice with concrete objects or structured investigations of a specific aspect of the learner's environment.

However, the reality is that far too little of current educational software makes effective use of the unique record-keeping, analysis, graphics, and feedback capabilities of the computer. For example, of the over 5,000 entries on EPIE's database of instructional and administrative educational software as of December 1984, only 5.4% were identified as "Simulation" (the type of software generally acknowledged as best exploiting the capabilities of computers), whereas 49.4% were identified as "drill-and-practice," with many of these failing to fully utilize the computer's capacity to provide and use feedback effectively.

SOFTWARE BY TYPE

Source: The Educational Software Selector (TESS), EPIE Institute. Teachers College Press, 1984.

Drill-and-practice	49.4%
Tutorial	18.9%
Gaming	12.3%
Simulation	5.4%
Computation Support	5.2%
Teacher Aid	2.2%
Other	6.6%

FEEDBACK AS A UNIQUE ATTRIBUTE OF COMPUTER SOFTWARE

The attribute of feedback and the analysis-of-feedback (which implies effective cumulative record-keeping of learner performance) to facilitate learning is at the base of the uniqueness of educational software. This attribute, however, is not as basic to educational software (or software in general) as it is to the cybernetic core of the computer itself. This becomes apparent if we think of a learner who is simply "fooling around" with a computer as an astructional device in his/her environment. Even though the computer hasn't been intentionally programmed to teach a thing, the learner receives immediate feedback on every action taken, and may, in the end, learn a good deal about how the computer operates as a result of having received such feedback. Similarly, a learner who is using a computer language like LOGO, or word processing software such as Bank Street Writer or WordStar (none of which are intentional learning materials), is going to learn a good deal through the use of such a material because of the

constant, instantaneous feedback on every action taken, even though that feedback is not based on the analysis of his/her performance aimed at the achievement of a set of specific objectives.

Given this fact of cybernetic life, it seems reasonable to expect that all software that even so much as implies an intention to facilitate learning should, by design, make use of feedback and the research-based tenets of teaching and learning that are based upon its effective use: (1) the well-researched evidence that feedback to the learner greatly facilitates learning (Thorndike et al.), (2) the growing body of research on the importance of adapting instructional strategies to the capabilities of the learners on the basis of feedback from learners (Nathenson and Henderson 1980), and (3), the commonsense evidence that tells us that a teacher's ability to make appropriate instructional management decisions depends on accurate feedback on learner performance. Not to employ these three important uses of feedback, in the face of such evidence, means failing to make use of the unique, built-in capacity of computer technology as applied to learning materials. It would be comparable to using the technology of the motion picture to portray only static scenes. If developers of educational software do, by design, fully exploit the potential of the feedback-based capacity of the computer, they may confirm another unique potential of educational software: the capability for facilitating the development of learning materials of the sort categorized as "Instructuring" on the Learning Materials Continuum.

Aside from the task of designing such materials, it also remains to be seen whether there is a sufficient market for learning materials. In other words, even if such materials were present in profusion, are schools ready to use

materials that enable learners to instructure their own learning? Whether educators will see such materials as being desireable is a policy decision that is yet to be made. But chances are, this decision is going to be made — consciously or by default — by educators and parents alike during the next few years as they respond to the evolving educational software market. Depending on the outcome of such de facto policy making there may or may not be a change in present practices of software development to fully exploit the capabilities of today's computers and instructional design technology.

In the meantime, the present market for instructionally unsophisticated drill-and-practice and tutorial software that fails to fully exploit the potential of the computer seems so large and so virile that software developers are hardly being pressed to move in the direction of creating more effective learning materials of an instructuring nature.

In other words, even though the technology exists through which such feedback-based, adaptive instructuring software can be created, the development of such unique self-instructional materials will depend to a great extent on the size of the educational market that opens up for such products. For the present, the instructuring software that does exist, exists largely within the military and industrial training sectors for which it has been developed — usually under a tightly-drawn proprietary cloak. No doubt there are those educators who believe that it should remain there. For those who may feel this way, the presence of such instructionally more sophisticated software in the average school curriculum would require a great many adjustments in the role of teachers, and in the role of learning materials.

The chances of the average school making such adjustments easily in the near future do not seem very great. This is one more reason why school and home consumers will continue to see a good deal more of the sort of software that currently dominates the market. As a major producer of educational software put it during a recent nationwide teleconference on the quality of educational software, "Producers will continue to develop what they are developing for two reasons: it's easy to produce and people buy it." This statement was quickly assented to by other participants across the country. (KETV/EPIE teleconference, "Quest for Quality: A National Dialogue on Educational Software")

This pithy identification of two major factors shaping the current educational software marketplace underlines the need for more discriminating educational consumers who are willing to resist purchasing software that is redundant of what's already out there. But given the pressure on schools to acquire more computers and the increasing numbers of parents who are buying a computer for their child's use at home, there will be continued purchasing of the drill-and-practice and unimaginative tutorial software that dominates the present market, and which, when judged by reasonable standards of quality, are barely acceptable.

To put this last statement in perspective, it should be noted that after a year and a half of evaluating hundreds of educational software products, EPIE Institute has found that only about one of every four products has been rated as meeting minimum standards of quality for both instructional and software design, and only three or four out of a hundred products have been judged to be of excellent quality in both these categories.

Because of the well-established practice of using workbooks that are presumed to help students practice skills previously taught from the textbook by the teacher,

schools are buying software that can be used to fit into that pattern. And they are not being very careful about the quality of what they are buying. Consequently, few schools are looking for more software that might be labeled "astructional," "structured," and "instructuring" on the Learning Materials Continuum. The reason for this may be that the use of astructional and structured materials may require too many decisions on the part of teachers, and instructuring materials (as they become more available) may be judged as not allowing for enough teacher decision-making. But the more important reason, no doubt, will be that all three types of material require a change from the established pattern of materials use in schools.

This means that most software purchased by schools (and by parents as well) is being viewed as being the electronic equivalent of the still-ubiquitous workbook -- or as an adjunct to the teacher-mediated textbook. As frequently as not this software is purchased without adequate evaluation and preview. The likelihood that this present pattern of software purchase will have a continuing depressing effect on the quality of educational software and its use is very great. This seems unavoidable when current sales patterns are clearly communicating to producers that current development practices are perfectly acceptable to consumers. This reinforces the production of software in a hurried "let's get it on the market" atmosphere that seldom involves the time during development to improve a product by using feedback from learners to shape its teaching effectiveness.

It is not surprising, therefore, to find (as in another paper in this Symposium) that few software producers are currently making learner feedback an essential part of the development process. Nor is this surprising to find that few schools are making evidence-of-learner-feedback-during-development (i.e., formative

evaluating and field testing) a major criterion in the selection of software. This is particularly unfortunate because, unlike traditional print and non-print materials, the electronic capabilities for analyzing and editing computer software make it easy to edit and improve based on feedback from learners and teachers.

Thus, despite the fact that research demonstrates — and common sense echoes — that feedback to learners, to teachers, and to developers (from learners and teachers) is important in the creation of effective, instructuring software, unfortunately, present practices and policies tend to lessen the likelihood that producers will undertake the development of such software.

Nevertheless, the potential of instructuring materials for the improvement of learning may be so great that once the efficacy of such materials becomes apparent, then current practices of in-school materials use may be significantly eroded. But no matter how impressive such evidence is, there are not likely to be any revolutionary changes in present patterns of materials use in many schools. I say this, not only because of the established strength of these patterns, but because it will take time for a significant flow of such materials to make their way onto the market.

However, if we assume that such a flow will be produced sooner rather than later it would seem most likely that the use of such materials will find their first niche within the school curriculum in areas in which there is less dependency on textbooks and in which teachers may be more open to the use of software that would enable learners to learn on their own.

If the above speculation has any validity, then I will suggest that the first curriculum area in which instructuring software may find significant use could be music. The reason this may prove to be the case is, first and foremost, the current excitement among both musicians and music educators being prompted by the increasingly sophisticated, yet easy to use, computer software now entering the market for teaching and composing music. One of the factors in stimulating the development of such software is the recent development of a standard computer interface (Musical Instrument Digital Interface) that establishes a hardware-software standard for interfacing a microcomputer with a great variety of previously incompatable electronic instruments. Given today's youngsters' great interest in both computers and electronic instruments, and the power of the music teaching software currently being produced by major software developers, many schools may find themselves making such instructuring software the basis of an individualized music curriculum before they have either access to or teacher readiness for such software in other curriculum areas. Nevertheless, other curriculum areas that are likely candidates are the higher levels of mathematics and science, which are national educational priorities for which schools are not likely to be able to enlist sufficient numbers of qualified teachers.

Still another possibility for the use of instructuring software is any curriculum area in which school administrations call for instructional accountability from teachers. When such accountability coincides with the availability of proven instructuring software, teachers may welcome the use of such software as accountability "insurance."

If and when instructuring software does become available in quantity and is accepted by teachers, its use is very likely to heighten the attention paid to what David Berliner has recently referred to as "the executive functions of

teaching" (Berliner 1983). Berliner describes the executive functions of teaching as: communicating goals; regulating, pacing and sequencing learning activities; controlling work time; running an orderly academic workplace; creating a pleasant work environment; and motivating and evaluating those being supervised.

The possibility that instructuring software will replace the teacher's traditional role as instructor with such executive functions (coupled with the students' use of structured software to explore and create on their own once teachers understand the potential of such materials) could, in time, lead to new attention to a major educational function of teaching that few teachers today have the time and training to do effectively: helping students see the interrelationship of all the things they are learning both in and outside of school. Were this important educational function to be accomplished, it might be the major positive result of the evaluation of educational software of an instructuring nature.

It would seem that if educators and parents are going to have access to the best software that the evolving technology of educational software is capable of producing, they are going to have to figure out how they can better work together to shape the large contextual social and economic factors that will ultimately control the educational quality of the technology. Creative, cooperative alliances of educators and parents that go beyond the buying of more hardware for school and home use are critically needed. Today such alliances exist in only a few communities. More need to be developed.

REFERENCES

EPIE Institute. The Educational Software Selector (TESS). New York. Teachers College Press, 1984

EPIE Institute. "Reading Materials: A-Structional and Instructional." EPIE REPORT No. 62/63, Selecting and Evaluating Beginning Reading Materials, Vol. VII, pp. 74-82. New York, May/June 1974.

Komoski, P. K. "An Imbalance of Product Quality and Instructional Quality: The Imperative of Empiricism." AV Communication Review, Volume 22, Number 4. Winter 1974.

Nathensen, M. B., and Henderson, E. S. Using Student Feedback to Improve Learning Materials. London, Croom Helm, 1980.

Thorndike, E. L. Educational Psychology. New York. Teachers College, Columbia University, 1913.

Wiener, Norbert. Cybernetics, 2nd edition. Cambridge. MIT Press, 1961.